

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Выборнова Любовь Алексеевна

Должность: Декан

Дата подписания: 03.02.2022 15:17:47

Уникальный программный код:

c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СЕРВИСА»
(ФГБОУ ВО «ПВГУС»)

Кафедра «Прикладная информатика в экономике»

РАБОЧАЯ УЧЕБНАЯ ПРОГРАММА

по дисциплине «Программирование»
для студентов специальности 09.02.05 «Прикладная информатика (по отраслям)»

Тольятти 2018 г.

Рабочая учебная программа по дисциплине «Программирование» включена в основную профессиональную образовательную программу специальности 09.02.05 «Прикладная информатика (по отраслям)» решением Президиума Ученого совета (Протокол № 4 от 28.06.2018 г.).

Начальник учебно-методического отдела _____
28.06.2018 г.



Н.М. Шемендюк

Рабочая учебная программа по дисциплине «Программирование» разработана в соответствии с Федеральным государственным образовательным стандартом специальности 09.02.05 «Прикладная информатика (по отраслям)», утвержденным приказом Минобрнауки России от 13.08.2014 г. № 1001.

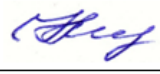
Составила Любивая Т.Г.

Согласовано Директор научной библиотеки  В.Н. Еремина

Согласовано Начальник управления информатизации  В.В. Обухов

Рабочая программа утверждена на заседании кафедры «Прикладная информатика в экономике»
Протокол № 12 от 22.06.2018 г.

Заведующий кафедрой  д.э.н., профессор Бердников В.А.

Согласовано Начальник учебно-методического отдела  Н.М. Шемендюк

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

1.1. Цели освоения дисциплины

Целями освоения дисциплины являются:

- изучение современных технологий программирования;
- формирование практических навыков разработки программных продуктов для решения профессиональных задач.

1.2. В соответствии с видами профессиональной деятельности, на которые ориентирована образовательная программа указанной специальности, содержание дисциплины позволит обучающимся решать следующие профессиональные задачи:

- разрабатывать программное обеспечение со статическим и динамическим контентом на основе готовых спецификаций и стандартов;
- выполнять отладку и тестирование программного обеспечения.

1.3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины у обучающихся формируются следующие компетенции:

Код компетенции	Наименование компетенции
ПК 2.2	Разрабатывать и публиковать программное обеспечение и информационные ресурсы отраслевой направленности со статическим и динамическим контентом на основе готовых спецификаций и стандартов.

1.4. Перечень планируемых результатов обучения по дисциплине

Результаты освоения дисциплины	Технологии формирования компетенции по указанным результатам	Средства и технологии оценки по указанным результатам
Знает: основы программирования (ПК 2.2).	Лекции	Устный опрос
Умеет: кодировать на языках программирования (ПК 2.2).	Лабораторные работы	Защита отчётов по лабораторным работам
Имеет практический опыт: разработки прикладного программного обеспечения (ПК 2.2).	Практические занятия	Защита отчётов по практическим занятиям

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к вариативной части.

Её освоение осуществляется в 6 и 7 семестрах* у студентов очной формы обучения, в 7 семестре* у студентов заочной формы обучения.

№ п/п	Наименование дисциплин, определяющих междисциплинарные связи	Код компетенции(й)
	<i>Предшествующие дисциплины</i>	
1.	Информатика и ИКТ	ОК 1-9
	<i>Последующие дисциплины</i>	
1.	Учебная практика	ОК 1-9, ПК 1.1-1.5, ПК 2.1-2.6

* Здесь и далее семестры указаны для обучающихся на базе основного общего образования. Для лиц, обучающихся на базе среднего общего образования, семестры соответствуют учебному плану и нормативному сроку обучения, установленному ФГОС.

3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Распределение фонда времени по семестрам и видам занятий

Виды занятий	очная форма обучения	очно-заочная форма обучения	заочная форма обучения
6 семестр			
Итого часов	153 ч.	-	-
Лекции (час)	50 ч.	-	-
Практические занятия (час)	46 ч.	-	-
Лабораторные работы (час)	22 ч.	-	-
Самостоятельная работа (час)	35 ч.	-	-
Курсовой проект (работа) (+,-)	-	-	-
Контрольная работа (+,-)	-	-	-
Экзамен, семестр/час.	-	-	-
Зачет, семестр	6 семестр	-	-
Контрольная работа, семестр	-	-	-
7 семестр			
Итого часов	105 ч.	-	258 ч.
Лекции (час)	22 ч.	-	8 ч.
Практические занятия (час)	28 ч.	-	8 ч.
Лабораторные работы (час)	20 ч.	-	6 ч.
Самостоятельная работа (час)	35 ч.	-	236 ч.
Курсовой проект (работа) (+,-)	-	-	-
Контрольная работа (+,-)	-	-	-
Экзамен, семестр/час	-	-	-
Дифференцированный зачет, семестр	7 семестр	-	7 семестр
Контрольная работа, семестр	-	-	-

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Содержание дисциплины

№ п/п	Раздел дисциплины	Виды учебных занятий, включая самостоятельную работу студентов и трудоемкость (в академических часах)				Средства и технологии оценки
		Лекции, час	Практические занятия, час	Лабораторные работы, час	Самостоятельная работа, час	
очная форма обучения						
6 семестр						
1.	Тема 1. Структурное программирование. Основное содержание: 1. Основные принципы структурного программирования. 2. Базовые конструкции структурного программирования.	12	-	-	8	Устный опрос
2.	Тема 2. Базовые средства языка C++. Основное содержание: 1. Типы данных C++. 2. Структура программы на языке C++. 3. Переменные и выражения. 4. Операторы языка, реализующие базовые конструкции структурного программирования. 5. Указатели и массивы. 6. Типы данных, определяемые пользователем.	12	30	14	8	Устный опрос, защита лабораторных и практических работ
3.	Тема 3. Модульное программирование. Основное содержание: 1. Функции. 2. Директивы препроцессора. 3. Области действия идентификаторов.	12	8	4	8	Устный опрос, защита лабораторных и практических работ
4.	Тема 4. Динамические структуры данных. Основное содержание: 1. Линейные списки. 2. Стеки. 3. Очереди. 4. Бинарные деревья.	14	8	4	11	Устный опрос, защита лабораторных и практических работ
	Промежуточная аттестация по дисциплине	50	46	22	35	Зачет

7 семестр						
5.	Тема 5. Объектно-ориентированное программирование. Основное содержание: 1. Логика объектно-ориентированного подхода. 2. Объектно-ориентированные языки программирования.	4	-	-	7	Устный опрос
6.	Тема 6. Классы. Основное содержание: 1. Описание класса. 2. Описание объектов. 3. Указатель this. 4. Конструкторы. 5. Статические элементы класса. 6. Дружественные функции и классы. 7. Деструкторы. 8. Перегрузка операций. 9. Указатели на элементы классов.	4	8	4	7	Устный опрос, защита лабораторных и практических работ
7.	Тема 7. Наследование. Основное содержание: 1. Виды наследования. 2. Простое наследование. 3. Виртуальные методы. 4. Множественное наследование. 5. Отличия структур и объединений от классов.	4	8	4	7	Устный опрос, защита лабораторных и практических работ
8.	Тема 8. Шаблоны классов. Основное содержание: 1. Создание шаблонов классов. 2. Использование шаблонов классов. 3. Специализация шаблонов классов. 4. Достоинства и недостатки шаблонов.	4	4	4	7	Устный опрос, защита лабораторных и практических работ
9.	Тема 9. Потокосовые классы. Основное содержание: 1. Стандартные потоки. 2. Методы обмена с потоками. 3. Файловые потоки. 4. Строковые потоки. 5. Потоки и типы, определенные пользователем.	6	8	8	7	Устный опрос, защита лабораторных и практических работ
	Промежуточная аттестация по дисциплине	22	28	20	35	Дифференцированный зачет
заочная форма обучения						

7 семестр						
1.	Тема 1. Структурное программирование. Основное содержание: 1. Основные принципы структурного программирования. 2. Базовые конструкции структурного программирования.	1	-	-	30	Устный опрос
2.	Тема 2. Базовые средства языка C++. Основное содержание: 1. Типы данных C++. 2. Структура программы на языке C++. 3. Переменные и выражения. 4. Операторы языка, реализующие базовые конструкции структурного программирования. 5. Указатели и массивы. 6. Типы данных, определяемые пользователем.	1	2,5	2	30	Устный опрос, защита лабораторных и практических работ
3.	Тема 3. Модульное программирование. Основное содержание: 1. Функции. 2. Директивы препроцессора. 3. Области действия идентификаторов.	1	0,5	1	30	Устный опрос, защита лабораторных и практических работ
4.	Тема 4. Динамические структуры данных. Основное содержание: 1. Линейные списки. 2. Стеки. 3. Очереди. 4. Бинарные деревья.	1	1	1	30	Устный опрос, защита лабораторных и практических работ
5.	Тема 5. Объектно-ориентированное программирование. Основное содержание: 1. Логика объектно-ориентированного подхода. 2. Объектно-ориентированные языки программирования.	1	-	-	30	Устный опрос, защита практических работ
6.	Тема 6. Классы. Основное содержание: 1. Описание класса. 2. Описание объектов. 3. Указатель this. 4. Конструкторы. 5. Статические элементы	1	1	0,5	30	Устный опрос, защита лабораторных и практических работ

	класса. 6. Дружественные функции и классы. 7. Деструкторы. 8. Перегрузка операций. 9. Указатели на элементы классов.					
7.	Тема 7. Наследование. Основное содержание: 1. Виды наследования. 2. Простое наследование. 3. Виртуальные методы. 4. Множественное наследование. 5. Отличия структур и объединений от классов.	1	1	0,5	30	Устный опрос, защита лабораторных и практических работ
8.	Тема 8. Шаблоны классов. Основное содержание: 1. Создание шаблонов классов. 2. Использование шаблонов классов. 3. Специализация шаблонов классов. 4. Достоинства и недостатки шаблонов.	0,5	1	-	12	Устный опрос, защита лабораторных и практических работ
9.	Тема 9. Потокосовые классы. Основное содержание: 1. Стандартные потоки. 2. Методы обмена с потоками. 3. Файловые потоки. 4. Строковые потоки. 5. Потоки и типы, определенные пользователем.	0,5	1	1	14	Устный опрос, защита лабораторных и практических работ
	Промежуточная аттестация по дисциплине	8	8	6	236	Дифференцированный зачет

4.2. Содержание практических занятий

№	Наименование темы практических занятий	Объем часов	Наименование темы дисциплины
очная форма обучения			
6 семестр			
1.	Практическое занятие 1. «Программирование алгоритмов линейной, разветвляющейся и циклической структур»	6	Базовые средства языка C++
2.	Практическое занятие 2. «Одномерные массивы и указатели»	4	Базовые средства языка C++
3.	Практическое занятие 3. «Двумерные массивы»	4	Базовые средства языка C++
4.	Практическое занятие 4. «Строки и файлы»	8	Базовые средства языка C++
5.	Практическое занятие 5. «Структуры»	8	Базовые средства языка C++
6.	Практическое занятие 6. «Функции»	8	Модульное программирование

7.	Практическое занятие 7. «Динамические структуры данных»	8	Динамические структуры данных
Итого за 6 семестр		46	
7 семестр			
8.	Практическое занятие 8. «Классы»	8	Классы
9.	Практическое занятие 9. «Наследование»	8	Наследование
10.	Практическое занятие 10. «Шаблоны классов»	4	Шаблоны классов
11.	Практическое занятие 11. «Стандартные потоки»	4	Потоковые классы
12.	Практическое занятие 12. «Файловые и строковые потоки»	4	Потоковые классы
Итого за 7 семестр		28	
Итого		74	
заочная форма обучения			
7 семестр			
1.	Практическое занятие 1. «Программирование алгоритмов линейной, разветвляющейся и циклической структур»	0,5	Базовые средства языка C++
2.	Практическое занятие 2. «Одномерные массивы и указатели»	0,5	Базовые средства языка C++
3.	Практическое занятие 3. «Двумерные массивы»	0,5	Базовые средства языка C++
4.	Практическое занятие 4. «Строки и файлы»	0,5	Базовые средства языка C++
5.	Практическое занятие 5. «Структуры»	0,5	Базовые средства языка C++
6.	Практическое занятие 6. «Функции»	0,5	Модульное программирование
7.	Практическое занятие 7. «Динамические структуры данных»	1	Динамические структуры данных
8.	Практическое занятие 8. «Классы»	1	Классы
9.	Практическое занятие 9. «Наследование»	1	Наследование
10.	Практическое занятие 10. «Шаблоны классов»	1	Шаблоны классов
11.	Практическое занятие 11. «Стандартные потоки»	1	Потоковые классы
Итого за 7 семестр		8	
Итого		8	

4.3. Содержание лабораторных работ

№	Наименование лабораторных работ	Объем часов	Наименование темы дисциплины
очная форма обучения			
6 семестр			
1.	Лабораторная работа 1. «Программирование алгоритмов линейной структуры»	2	Базовые средства языка C++
2.	Лабораторная работа 2. «Программирование алгоритмов разветвляющейся структуры»	2	Базовые средства языка C++
3.	Лабораторная работа 3. «Программирование алгоритмов циклической структуры»	2	Базовые средства языка C++

4.	Лабораторная работа 4. «Одномерные и двумерные массивы»	4	Базовые средства языка C++
5.	Лабораторная работа 5. «Структуры»	4	Базовые средства языка C++
6.	Лабораторная работа 6. «Функции и файлы»	4	Модульное программирование
7.	Лабораторная работа 7. «Динамические структуры данных»	4	Динамические структуры данных
	Итого за 6 семестр	22	
7 семестр			
8.	Лабораторная работа 8. «Классы»	4	Классы
9.	Лабораторная работа 9. «Наследование»	4	Наследование
10.	Лабораторная работа 10. «Шаблоны классов»	4	Шаблоны классов
11.	Лабораторная работа 11. «Стандартные потоки»	4	Потоковые классы
12.	Лабораторная работа 12. «Файловые и строковые потоки»	4	Потоковые классы
	Итого за 7 семестр	20	
	Итого	42	
заочная форма обучения			
7 семестр			
1.	Лабораторная работа 1. «Программирование алгоритмов циклической структуры»	0,5	Базовые средства языка C++
2.	Лабораторная работа 2. «Одномерные и двумерные массивы»	0,5	Базовые средства языка C++
3.	Лабораторная работа 3. «Структуры»	1	Базовые средства языка C++
4.	Лабораторная работа 4. «Функции и файлы»	1	Модульное программирование
5.	Лабораторная работа 5. «Динамические структуры данных»	1	Динамические структуры данных
6.	Лабораторная работа 6. «Классы»	0,5	Классы
7.	Лабораторная работа 7. «Наследование»	0,5	Наследование
8.	Лабораторная работа 8. «Стандартные потоки»	1	Потоковые классы
	Итого за 7 семестр	6	
	Итого	6	

5. Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине

Технологическая карта самостоятельной работы студента

Код реализуемой компетенции	Вид деятельности студентов (задания на самостоятельную работу)	Итоговый продукт самостоятельной работы	Средства и технологии оценки	Объем часов
ПК 2.2	Ответы на контрольные вопросы	Конспект	Тестирование	35/-/-
Итого за 6 семестр				35/-/-
ПК 2.2	Работа с литературой	Конспект	Тестирование	35/-/236
Итого за 7 семестр				35/-/236
Итого				70/-/236

Рекомендуемая литература: 1, 2, 3, 4, 5, 6, 7, 8.

Содержание заданий для самостоятельной работы

Вопросы для самоконтроля

1. Основные принципы структурного программирования.
2. Базовые конструкции структурного программирования.
3. Типы данных C++.
4. Структура программы на языке C++.
5. Переменные и выражения.
6. Оператор «выражение».
7. Операторы ветвления.
8. Операторы цикла.
9. Операторы передачи управления.
10. Указатель на функцию.
11. Указатель на объект.
12. Указатель на void.
13. Операции с указателями.
14. Ссылки.
15. Массивы.
16. Переименование типов (typedef).
17. Перечисления (enum).
18. Структуры (struct).
19. Объединения (union).
20. Объявление и определение функций.
21. Глобальные переменные.
22. Параметры функции.
23. Рекурсивные функции.
24. Шаблоны функций.
25. Функция main().
26. Функции стандартной библиотеки.
27. Директива #include.
28. Директива #define.
29. Директивы условной компиляции.
30. Директива #undef.
31. Области действия идентификаторов.
32. Динамические структуры данных.
33. Основные принципы объектно-ориентированного программирования (ООП).
34. Основные свойства ООП.
35. Объектно-ориентированные языки программирования.
36. Описание класса.
37. Описание объектов.
38. Указатель this.
39. Свойства конструкторов.
40. Статические элементы класса.
41. Дружественные функции и классы.
42. Деструкторы.
43. Перегрузка операций.
44. Указатель на метод класса.
45. Указатель на поле класса.
46. Виды наследования.
47. Простое наследование.
48. Множественное наследование.
49. Отличия структур и объединений от классов.

50. Правила описания шаблонов.
51. Использование шаблонов классов.
52. Достоинства и недостатки шаблонов.
53. Стандартные потоки.
54. Методы обмена с потоками.
55. Файловые потоки.
56. Строковые потоки.
57. Потоки и типы, определенные пользователем.

6. Методические указания для обучающихся по освоению дисциплины

Инновационные образовательные технологии

Вид образовательных технологий, средств передачи знаний, формирования умений и практического опыта	№ темы/тема лекции	№ практического (семинарского) занятия/наименование темы	№ лабораторной работы/цель
Слайд-лекция	Тема 4. Динамические структуры данных.		

В начале семестра студентам необходимо ознакомиться с технологической картой дисциплины, выяснить, какие результаты освоения дисциплины заявлены (знания, умения, практический опыт). Для успешного освоения дисциплины студентам необходимо выполнить задания, предусмотренные рабочей учебной программой дисциплины и пройти контрольные точки в сроки, указанные в технологической карте (раздел 11). От качества и полноты их выполнения будет зависеть уровень сформированности компетенций и оценка текущей успеваемости по дисциплине. По итогам текущей успеваемости студенту может быть выставлена оценка по промежуточной аттестации. Списки учебных пособий, научных трудов, которые студентам следует прочесть и законспектировать, темы лабораторных работ, вопросы к зачету (дифференцированному зачету) и другие необходимые материалы указаны в разработанном для данной дисциплины учебно-методическом пособии.

Основной формой освоения дисциплины является контактная работа с преподавателем – лекции, лабораторные работы, консультации, в том числе проводимые с применением дистанционных технологий.

По дисциплине часть тем изучается студентами самостоятельно. Самостоятельная работа предусматривает подготовку к аудиторным занятиям, выполнение заданий, подготовку к промежуточной аттестации.

На лекционных и практических занятиях вырабатываются навыки и умения обучающихся по применению полученных знаний в конкретных ситуациях, связанных с будущей профессиональной деятельностью. По окончании изучения дисциплины проводится промежуточная аттестация (дифференцированный зачет).

Регулярное посещение аудиторных занятий не только способствует успешному овладению знаниями, но и помогает организовать время, т.к. все виды учебных занятий распределены в семестре планомерно, с учетом необходимых временных затрат.

6.1. Методические указания для обучающихся по освоению дисциплины на практических занятиях, лабораторных работах

Практические занятия обучающихся обеспечивают:

- проверку и уточнение знаний, полученных на лекциях;
- получение навыков составления докладов и сообщений, обсуждения вопросов по учебному материалу дисциплины;
- обсуждение вопросов в аудитории, разделенной на группы 6-8 обучающихся либо индивидуальных;

- выполнение практических заданий, задач;
- подведение итогов занятий по рейтинговой системе, согласно технологической карте дисциплины.

Содержание заданий для практических занятий

Практическое занятие 1. «Программирование алгоритмов линейной, разветвляющейся и циклической структур».

Вычислить значения функции F на интервале от x_n до x_k с шагом dx .

$$F = \begin{cases} ax^2 + \frac{b}{c} & \text{при } x < 1 \text{ и } c \neq 0, \\ \frac{x-a}{(x-c)^2} & \text{при } x > 1,5 \text{ и } c = 0, \\ \frac{x^2}{c^2} & \text{в остальных случаях,} \end{cases}$$

где a, b, c – действительные числа.

Практическое занятие 2. «Одномерные массивы и указатели».

В одномерном массиве, состоящем из n вещественных элементов, вычислить количество отрицательных элементов массива.

Упорядочить элементы массива по возрастанию.

Практическое занятие 3. «Двумерные массивы».

Дана целочисленная прямоугольная матрица. Определить количество строк, не содержащих ни одного нулевого элемента.

Практическое занятие 4. «Строки и файлы».

Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие введенное с клавиатуры слово.

Практическое занятие 5. «Структуры».

Описать структуру с именем WORKER, содержащую следующие поля:

- фамилия и инициалы работника;
- название занимаемой должности;
- год поступления на работу.

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из десяти структур типа WORKER; записи должны быть размещены по алфавиту;
- вывод на дисплей фамилий работников, чей стаж работы в организации превышает значение, введенное с клавиатуры;
- если таких работников нет, вывести соответствующее сообщение.

Практическое занятие 6. «Функции».

Написать программу упорядочивания массива методом быстрой сортировки, используя рекурсию.

Практическое занятие 7. «Динамические структуры данных».

Создать программу, отыскивающую проход по лабиринту. Лабиринт представляется в виде матрицы, состоящей из квадратов. Каждый квадрат либо открыт, либо закрыт. Вход в закрытый квадрат запрещен. Если квадрат открыт, то вход в него возможен со стороны, но не с угла. Каждый квадрат определяется его координатами в матрице.

Программа находит проход через лабиринт, двигаясь от заданного входа. После отыскания прохода программа выводит найденный путь в виде координат квадратов. Для хранения пути использовать стек.

Практическое занятие 8. «Классы».

Построить систему классов для описания плоских геометрических фигур: круг, квадрат, прямоугольник. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и вращения на заданный угол.

Написать программу, демонстрирующую работу с этими классами. Программа должна содержать меню, позволяющее осуществить проверку всех методов классов.

Практическое занятие 9. «Наследование».

Разработать программу, имитирующую работу функционального калькулятора, который позволяет выбрать с помощью меню какую-либо из известных ему функций, затем предлагает ввести значение аргумента и, возможно, коэффициентов и после ввода выдает соответствующее значение функции.

Практическое занятие 10. «Шаблоны классов».

Обработка исключительных ситуаций». Разработать шаблонный класс для представления разреженных одномерных массивов. Размер логического массива передавать через аргумент конструктора. Класс должен обеспечивать хранение данных любого типа T, для которого предусмотрены конструктор по умолчанию, конструктор копирования и операция присваивания. Класс должен содержать операцию индексирования, возвращающую ссылку на найденный элемент в массиве. Если элемент с заданным индексом не найден, то операция должна создать новый элемент с этим индексом и поместить его в массив.

При необходимости добавить в класс другие методы. В клиенте main() продемонстрировать использование этого класса.

Практическое занятие 11. «Стандартные потоки».

Определить класс с именем WORKER, содержащий следующие поля:

- фамилия и инициалы работника;
- название занимаемой должности;
- год поступления на работу.

Определить методы доступа к этим полям и перегруженные операции извлечения и вставки для объектов типа WORKER.

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из десяти объектов типа WORKER; записи должны быть размещены по алфавиту;
- вывод на дисплей фамилий работников, чей стаж работы в организации превышает значение, введенное с клавиатуры;
- если таких работников нет, вывести на дисплей соответствующее сообщение.

Практическое занятие 12. «Файловые и строковые потоки».

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

Лабораторные работы

№	Наименование лабораторной работы	Задания по лабораторной работе
1.	Лабораторная работа 1. «Программирование алгоритмов линейной структуры».	Вычислить объем пирамиды, в основании которой лежит прямоугольник. Стороны прямоугольника равны: $a=6\text{см}$, $b=4\text{см}$, высота $h=10\text{см}$. Формулы: $S_{\text{осн}} = ab$, $V = \frac{1}{3} S_{\text{осн}} h$.
2.	Лабораторная работа 2. «Программирование алгоритмов разветвляющейся структуры».	Вычислить значение функции:

		$F = \begin{cases} \frac{1}{ax} - b & \text{при } x + 5 < 0 \text{ и } c = 0, \\ \frac{x - a}{x} & \text{при } x + 5 > 0 \text{ и } c \neq 0, \\ \frac{10x}{c - 4} & \text{в остальных случаях,} \end{cases}$ где a, b, c – действительные числа.
3.	Лабораторная работа 3. «Программирование алгоритмов циклической структуры».	Вычислить и вывести на экран в виде таблицы значения функции, заданной с помощью ряда Тейлора, на интервале от $x_{\text{нач}}$ до $x_{\text{кон}}$ с шагом dx с точностью ε: $\ln(x + 1) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{n+1}}{n + 1} =$ $= x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} - \dots, -1 < x < 1.$ Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.
4.	Лабораторная работа 4. «Одномерные и двумерные массивы».	1. В одномерном массиве, состоящем из n вещественных элементов, вычислить произведение элементов массива, расположенных между максимальным и минимальным элементами. Упорядочить элементы массива по возрастанию. 2. Дана целочисленная квадратная матрица. Определить максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.
5.	Лабораторная работа 5. «Структуры».	1. Описать структуру с именем STUDENT, содержащую следующие поля: ▪ фамилия и инициалы; ▪ номер группы; ▪ успеваемость (массив из пяти элементов). 2. Написать программу, выполняющую следующие действия: ▪ ввод с клавиатуры данных в массив, состоящий из десяти структур типа STUDENT; записи должны быть упорядочены по возрастанию номера группы; ▪ вывод на экран фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4,0; ▪ если таких студентов нет, вывести соответствующее сообщение.
6.	Лабораторная работа 6. «Функции и файлы».	Для хранения данных о планшетных сканерах описать структуру вида: <pre>struct scan_info{ char model[25]; // наименование модели int price; // цена double x_size; // горизонтальный размер области сканирования double y_size; // вертикальный размер области сканирования int optr; // оптическое разрешение</pre>

		int grey: // число градаций серого }; Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых двух байтах размещается значение типа int, определяющее количество сделанных в файле записей; далее без пропусков размещаются записи о сканерах. Написать функцию, которая извлекает из этого файла данные о сканере в структуру типа scan_info. Обязательный параметр – номер требуемой записи. Функция должна возвращать нулевое значение, если чтение прошло успешно, и -1 в противном случае. Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) – 6-8 записей и выводящей на экран данные о запрошенной записи. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.
7.	Лабораторная работа 7. «Динамические структуры данных».	Составить программу, которая содержит текущую информацию о книгах в библиотеке. Сведения о книгах содержат: ▪ номер УДК; ▪ фамилию и инициалы автора; ▪ название; ▪ год издания; ▪ количество экземпляров данной книги в библиотеке. Программа должна обеспечивать: ▪ начальное формирование данных о всех книгах в библиотеке в виде двоичного дерева; ▪ добавление данных о книгах, вновь поступающих в библиотеку; ▪ удаление данных о списываемых книгах; ▪ по запросу выдаются сведения о наличии книг в библиотеке, упорядоченные по годам издания.
8.	Лабораторная работа 8. «Классы».	Описать класс, реализующий бинарное дерево, обладающее возможностью добавления новых элементов, удаления существующих, поиска элемента по ключу, а также последовательного доступа ко всем элементам. Написать программу, использующую этот класс для представления англо-русского словаря. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса. Предусмотреть возможность формирования словаря из файла и с клавиатуры.
9.	Лабораторная работа 9. «Наследование».	Создать класс CPoint – точка. На его основе создать классы CcoloredPoint и CLine. На основе класса CLine создать класс CColoredLine и класс CPolyLine – многоугольник. Все классы должны

		иметь методы для установки и получения значений всех координат, а также изменения цвета и получения текущего цвета. Написать демонстрационную программу, в которой будет использоваться список объектов этих классов в динамической памяти.
10.	Лабораторная работа 10. «Шаблоны классов».	Создать шаблон класса «стек». Использовать его при решении задачи «Классы».
11.	Лабораторная работа 11. «Стандартные потоки».	Определить класс с именем STUDENT, содержащий следующие поля: ▪ фамилия и инициалы; ▪ номер группы; ▪ успеваемость (массив из пяти элементов). Определить методы доступа к этим полям и перегруженные операции извлечения и вставки для объектов типа STUDENT. 2. Написать программу, выполняющую следующие действия: ▪ ввод с клавиатуры данных в массив, состоящий из десяти объектов типа STUDENT; записи должны быть упорядочены по возрастанию номера группы; ▪ вывод на дисплей фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4.0; ▪ если таких студентов нет, вывести соответствующее сообщение.
12.	Лабораторная работа 12. «Файловые и строковые потоки».	Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие заданное с клавиатуры слово.

Лабораторные работы обеспечивают: демонстрацию применения теоретических знаний на практике, закрепление и углубление теоретических знаний, контроль знаний и умений в формулировании выводов, развитие интереса к изучаемой дисциплине.

Применение лабораторных работ позволяет вовлечь в активную работу всех обучающихся группы и сформировать интерес к изучению дисциплины.

Самостоятельный поиск ответов на поставленные вопросы и задачи в ходе лабораторной работы приобретают особую значимость в восприятии, понимании содержания дисциплины.

Изученный на лекциях материал лучше усваивается, лабораторные работы демонстрируют практическое их применение.

6.2. Методические указания для выполнения контрольных работ

Контрольная работа по дисциплине учебным планом не предусмотрена.

6.3. Методические указания для выполнения курсовых работ

Курсового проекта (работы) учебным планом не предусмотрено.

7. Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине

Фонды оценочных средств, позволяющие оценить уровень сформированности

компетенций и результаты освоения дисциплины, представлены следующими компонентами:

Код оцениваемой компетенции (или) её части	Тип контроля	Вид контроля	Количество элементов, шт.
очная форма обучения			
ПК 2.2	текущий	устный опрос	1-32 – 6 семестр 33-57 – 7 семестр
ПК 2.2	текущий	защита отчётов по лабораторным работам	7 – 6 семестр 5 – 7 семестр
ПК 2.2	текущий	защита отчётов по практическим занятиям	7 – 6 семестр 5 – 7 семестр
ПК 2.2	промежуточный	вопросы типа «Эссе»	30 – 6 семестр
ПК 2.2	промежуточный	компьютерный тест	80 – 7 семестр
заочная форма обучения			
ПК 2.2	текущий	устный опрос	57
ПК 2.2	текущий	защита отчётов по лабораторным работам	8
ПК 2.2	текущий	защита отчётов по практическим занятиям	11
ПК 2.2	промежуточный	компьютерный тест	80

7.1. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Результаты освоения дисциплины	Оценочные средства
ПК 2.2 Знает: основы программирования.	Ответить на вопросы: 1. Базовые конструкции структурного программирования. 2. Структура программы на языке C++. 3. Рекурсивные функции. 4. Основные свойства объектно-ориентированного программирования. 5. Правила описания шаблонов.
ПК 2.2 Умеет: кодировать на языках программирования.	<i>очная форма обучения</i> Лабораторная работа 1. «Программирование алгоритмов линейной структуры». Лабораторная работа 2. «Программирование алгоритмов разветвляющейся структуры». Лабораторная работа 3. «Программирование алгоритмов циклической структуры». Лабораторная работа 4. «Одномерные и двумерные массивы». Лабораторная работа 5. «Структуры». Лабораторная работа 6. «Функции и файлы». Лабораторная работа 7. «Динамические структуры данных».

	Лабораторная работа 8. «Классы». Лабораторная работа 9. «Наследование». Лабораторная работа 10. «Шаблоны классов». Лабораторная работа 11. «Стандартные потоки». Лабораторная работа 12. «Файловые и строковые потоки». <i>заочная форма обучения</i> Лабораторная работа 1. «Программирование алгоритмов циклической структуры». Лабораторная работа 2. «Одномерные и двумерные массивы». Лабораторная работа 3. «Структуры». Лабораторная работа 4. «Функции и файлы». Лабораторная работа 5. «Динамические структуры данных». Лабораторная работа 6. «Классы». Лабораторная работа 7. «Наследование». Лабораторная работа 8. «Стандартные потоки».
ПК 2.2 Имеет практический опыт: разработки прикладного программного обеспечения.	<i>очная форма обучения</i> Практическое занятие 1. «Программирование алгоритмов линейной, разветвляющейся и циклической структур». Практическое занятие 2. «Одномерные массивы и указатели». Практическое занятие 3. «Двумерные массивы». Практическое занятие 4. «Строки и файлы». Практическое занятие 5. «Структуры». Практическое занятие 6. «Функции». Практическое занятие 7. «Динамические структуры данных». Практическое занятие 8. «Классы». Практическое занятие 9. «Наследование». Практическое занятие 10. «Шаблоны классов». Практическое занятие 11. «Стандартные потоки». Практическое занятие 12. «Файловые и строковые потоки». <i>заочная форма обучения</i> Практическое занятие 1. «Программирование алгоритмов линейной, разветвляющейся и циклической структур». Практическое занятие 2. «Одномерные массивы и указатели». Практическое занятие 3. «Двумерные массивы». Практическое занятие 4. «Строки и файлы». Практическое занятие 5. «Структуры». Практическое занятие 6. «Функции». Практическое занятие 7. «Динамические структуры данных». Практическое занятие 8. «Классы». Практическое занятие 9. «Наследование». Практическое занятие 10. «Шаблоны классов». Практическое занятие 11. «Стандартные потоки».

7.2. Методические рекомендации к определению процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Рабочая учебная программа дисциплины содержит следующие структурные элементы:

- перечень компетенций, формируемых в результате изучения дисциплины в процессе освоения образовательной программы;
- типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности в процессе освоения образовательной программы (далее—задания). Задания по каждой компетенции, как правило, не должны повторяться.

Требования по формированию задания на оценку ЗНАНИЙ:

- обучающийся должен воспроизводить и объяснять учебный материал с требуемой степенью научной точности и полноты;
- применяются средства оценивания компетенций: тестирование, вопросы по основным понятиям дисциплины и т.п.

Требования по формированию задания на оценку УМЕНИЙ:

- обучающийся должен решать типовые задачи (выполнять задания) на основе воспроизведения стандартных алгоритмов решения;
- применяются следующие средства оценивания компетенций: простые ситуационные задачи (задания) с коротким ответом или простым действием, упражнения, задания на соответствие или на установление правильной последовательности, эссе и другое.

Требования по формированию задания на оценку навыков и (или) ОПЫТА ДЕЯТЕЛЬНОСТИ:

- обучающийся должен решать усложненные задачи (выполнять задания) на основе приобретенных знаний, умений и навыков, с их применением в определенных ситуациях;
- применяются средства оценивания компетенций: задания требующие многошаговых решений как в известной, так и в нестандартной ситуациях, задания, требующие поэтапного решения и развернутого ответа, ситуационные задачи, проектная деятельность, задания расчетно-графического типа. Средства оценивания компетенций выбираются в соответствии с заявленными результатами обучения по дисциплине.

Процедура выставления оценки доводится до сведения обучающихся в течение месяца с начала изучения дисциплины путем ознакомления их с технологической картой дисциплины, которая является неотъемлемой частью рабочей учебной программы по дисциплине.

В результате оценивания компетенций по дисциплине студенту начисляются баллы по шкале, указанной в рабочей учебной программе по дисциплине.

7.3. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Успешность усвоения дисциплины характеризуется качественной оценкой на основе листа оценки сформированности компетенций, который является приложением к зачетно-экзаменационной ведомости при проведении промежуточной аттестации по дисциплине.

Критерии оценивания компетенций

Компетенция считается сформированной, если теоретическое содержание курса освоено полностью; при устных собеседованиях студент исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 86 до 100, что соответствует *повышенному уровню сформированности компетенции*.

Компетенция считается сформированной, если теоретическое содержание курса освоено полностью; при устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и

другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 61 до 85,9, что соответствует *пороговому уровню* сформированности компетенции.

Компетенция считается несформированной, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет практические работы, не демонстрирует необходимых умений, доля невыполненных заданий, предусмотренных рабочей учебной программой составляет 55 %, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже 61, что соответствует *допороговому уровню*.

Шкала оценки уровня освоения дисциплины

Качественная оценка может быть выражена: в процентном отношении качества усвоения дисциплины, которая соответствует баллам, и переводится в уровневую шкалу и оценки «отлично» / 5, «хорошо» / 4, «удовлетворительно» / 3, «неудовлетворительно» / 2, «зачтено», «не зачтено». Преподаватель ведет письменный учет текущей успеваемости студента в соответствии с технологической картой по дисциплине.

Шкала оценки результатов освоения дисциплины, сформированности компетенций

Шкалы оценки уровня сформированности компетенции(й)		Шкала оценки уровня освоения дисциплины		
Уровневая шкала оценки компетенций	100-балльная шкала, %	100-балльная шкала, %	5-балльная шкала, дифференцированная оценка/балл	Недифференцированная оценка
допороговый	ниже 61	ниже 61	«неудовлетворительно» / 2	не зачтено
пороговый	61-85,9	70-85,9	«хорошо» / 4	зачтено
		61-69,9	«удовлетворительно» / 3	зачтено
повышенный	86-100	86-100	«отлично» / 5	зачтено

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Списки основной литературы

1. Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы [Электронный ресурс] : учеб. для студентов техн. специальностей / В. А. Гвоздева. - Документ Bookread2. - М. : ФОРУМ [и др.], 2015. - 541 с. - Режим доступа: <http://znanium.com/bookread2.php?book=492670#>.

2. Колдаев, В. Д. Основы алгоритмизации и программирования [Электронный ресурс] : учеб. пособие для сред. проф. образования по группе специальностей "Информатика и вычисл. техника" / В. Д. Колдаев под ред. Л. Г. Гагариной. - Документ Bookread2. - М. : ФОРУМ [и др.], 2015. - 413 с. : ил., табл. - Режим доступа: <http://znanium.com/bookread2.php?book=484837#>.

3. Кузин, А. В. Программирование на языке Си [Электронный ресурс] : [учеб. пособие для вузов] / А. В. Кузин, Е. В. Чумакова. - Документ Bookread2. - М. : ФОРУМ, 2015. - 142 с. - Режим доступа: <http://znanium.com/bookread2.php?book=505194>.

4. Учебно-методический комплекс по дисциплине "Программирование" [Электронный ресурс] : для студентов специальности 09.02.05 "Приклад. информатика (по отраслям)" / Поволж. гос. ун-т сервиса (ФГБОУ ВПО "ПВГУС"), Каф. "Приклад.

информатика в экономике" ; сост. В. С. Марченко. - Документ Adobe Acrobat. - Тольятти : ПВГУС, 2016. - 1,48 МБ, 96 с. : ил. - Режим доступа: <http://elib.tolgas.ru>.

Списки дополнительной литературы

5. Голицына, О. Л. Языки программирования [Электронный ресурс] : учеб. пособие для сред. проф. образования / О. Л. Голицына, Т. Л. Партыка, И. И. Попов. - 3-е изд., перераб. и доп. - Документ Bookread2. - М. : Форум [и др.], 2015. - 398 с. - Режим доступа: <http://znanium.com/bookread2.php?book=493421>.

6. Хорев, П. Б. Объектно-ориентированное программирование с примерами на С# [Электронный ресурс] : учеб. пособие для вузов по направлениям 01.03.02 "Приклад. математика и информатика" и 09.03.01 «Информатика и вычисл. техника»/ П. Б. Хорев. - М. : Форум, НИЦ ИНФРА-М, 2016. - 200 с. - Режим доступа: <http://znanium.com/bookread2.php?book=529350>.

7. Электронный учебный курс по дисциплине "Программирование" [Электронный ресурс] : для студентов направления подгот. 38.03.05 "Бизнес-информатика", 09.03.03 "Приклад. информатика" / Поволж. гос. ун-т сервиса (ФГБОУ ВПО "ПВГУС") ; сост. В. С. Марченко. - Документ Adobe Flash Player. - Тольятти : ПВГУС, 2015. - 46,9 МБ. - CD-ROM.

8. Яшин, В. Н. Информатика. Программные средства персонального компьютера [Электронный ресурс] : учеб. пособие для вузов по направлению "Приклад. информатика" и др. экон. специальностям / В. Н. Яшин. - Документ Bookread2. - М. : ИНФРА-М, 2018. - 236 с. - Режим доступа: <http://znanium.com/bookread2.php?book=937489>.

8.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее – сеть "Интернет"), необходимых для освоения дисциплины

Интернет-ресурсы

1. ИНТУИТ. Национальный открытый университет [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/>. – Загл. с экрана.

2. Российское образование [Электронный ресурс] : федер. портал. - Режим доступа: <http://www.edu.ru>. - Загл. с экрана.

3. Электронная библиотечная система Поволжского государственного университета сервиса [Электронный ресурс]. - Режим доступа: <http://elib.tolgas.ru/>. - Загл. с экрана.

4. Электронно-библиотечная система Znanium.com [Электронный ресурс]. - Режим доступа: <http://znanium.com/>. - Загл. с экрана.

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Краткая характеристика применяемого программного обеспечения

№ п/п	Программный продукт	Характеристика	Назначение при освоении дисциплины
1.	Microsoft Office	Пакет прикладных программ	Оформление отчетов
2.	Microsoft Visual Studio	Среда программирования	Выполнение лабораторных и практических работ

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Реализация программы дисциплины в соответствии с требованиями ФГОС СПО по специальности требует наличие учебного кабинета, укомплектованного

специализированной мебелью, техническими средствами обучения и наглядными пособиями, служащими для представления учебной информации.

Для реализации программы учебной дисциплины предусмотрены следующие специальные помещения:

для проведения занятий лекционного типа используются специальные помещения – учебные аудитории, оснащенные оборудованием, техническими средствами обучения, служащими для представления учебной информации;

для проведения занятий семинарского типа (*лабораторных работ, практических занятий*), групповых и индивидуальных консультаций используются специальные помещения – учебные аудитории, оснащенные оборудованием, техническими средствами обучения и материалами, учитывающими требования международных стандартов;

для текущего контроля и промежуточной аттестации используются специальные помещения – учебные аудитории, укомплектованные специализированной мебелью, и (или) компьютерные классы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета;

для самостоятельной работы обучающихся используются специальные помещения – учебные аудитории для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета.

11. Примерная технологическая карта дисциплины «Программирование»

кафедра «Прикладная информатика в экономике»

преподаватель _____, специальность 09.02.05 «Прикладная информатика (по отраслям)»

№	Виды контрольных точек	Количество контрольных точек	Количество баллов за 1 контрольную точку	Срок прохождения контрольных точек																Итого	Зачетно-экзаменационная сессия	
				Февраль				Март				Апрель				Май						
1.	Обязательные задания:																					
1.1.	Выполнение лабораторных работ	7	5			+		+		+		+		+		+			35			
1.2.	Выполнение практических работ	7	5		+		+		+		+		+		+		+		35			
2.	Дополнительные задания																					
2.1.	Итоговое тестирование	1	30														+		30			
	Общий рейтинг по дисциплине:																		100			
	Форма контроля																			Зачет		

кафедра «Прикладная информатика в экономике»
преподаватель _____, специальность 09.02.05 «Прикладная информатика (по отраслям)»

№	Виды контрольных точек	Количество контрольных точек	Количество баллов за 1 контрольную точку	Срок прохождения контрольных точек																Итого	Зачетно-экзаменационная сессия	
				Сентябрь				Октябрь				Ноябрь				Декабрь						
1.	Обязательные задания:																					
1.1.	Выполнение лабораторных работ	5	7					+		+		+		+					35			
1.2.	Выполнение практических работ	5	7				+		+		+		+						35			
2.	Дополнительные задания																					
2.1.	Итоговое тестирование	1	30												+				30			
	Общий рейтинг по дисциплине:																		100			
	Форма контроля																			Дифференцированный зачет		

